UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/972,010	10/05/2001	Duane Mark Baldwin	SJ09-2001-0093	4421

46917 7590 07/14/2006

KONRAD RAYNES & VICTOR, LLP.
ATTN: IBM37
315 SOUTH BEVERLY DRIVE, SUITE 210
BEVERLY HILLS, CA 90212

EXAMINER	
CHOUDHURY, AZIZUL Q	
ART UNIT	PAPER NUMBER
2145	

DATE MAILED: 07/14/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/972,010	BALDWIN ET AL.	
	Examiner	Art Unit	
	Azizul Choudhury	2145	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *25 April 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-21 and 23-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3-21 and 23-26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *05 October 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Detailed Action

This office action is in response to the correspondence received on April 25, 2006.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1, 3-21 and 23-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bates et al (US Pat No: US006977927B1) in view of Phillips et al (US Pat No: US005321828A), hereafter referred to as Bates and Phillips, respectively.

1. With regards to claims 1, 15 and 21, Bates teaches through Phillips, a storage area network (SAN) having first and second digital data processors and one or more storage devices in communication with the digital data processors comprising:
   - A manager in communication with the SAN components (Figure 7, element 104, Bates);
   - A first platform-specific process executing on the first digital data processor, the first digital data processor executing under a first operating system (Figure 7, element 702, Bates);

- A second platform-specific process executing on the second digital data processor, the second digital data processor executing under a second operating system different from the first platform.(Figure 7, element 704, Bates);
- A first common platform-independent process executing on the first digital data processor, wherein the first common platform independent process invokes and communicates with a second command line interface of the second operating system to effect execution of the first platform-specific process via one or more command-line parameters (GDB communicates with Windows through ASCII strings in a command line interface (column 23, lines 50-67, Phillips));
- A second common platform-independent process executing on the second digital data processor, wherein the second common platform independent process invokes and communicates with a second command line interface of the second operating system to effect execution of the first platform-specific process via one or more command-line parameters (GDB communicates with Windows through ASCII strings in a command line interface (column 23, lines 50-67, Phillips). Furthermore, communication of applications through ASCII is well known in the art to incorporate command lines and when command line arguments are transferred between applications, it is well known that the option to leave the command line window open is available); and

- The manager transmits a query to the first and second common platform-independent processes to request information regarding one or more of the SAN components and the platform independent processes invoke the first and second platform-specific processes, respectively, to obtain the requested information (Storage allocator communicates with servers and storage to process platform-specific processes (column 13, line 29- column 14, line 60, Bates).

  While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

  Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates).

2. With regards to claim 3, Bates teaches through Phillips, the SAN, wherein each of the first and the second operating systems can be any of a Unix™, a Windows™, Solaris, AIX operating systems

(Bates' design allows for NT (Windows) and Unix (Figure 7, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Phillips' design also permits the use of Windows and Unix (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

3. With regards to claims 4, 16 and 23, Bates teaches through Phillips, the SAN, comprising a manager in communication with the first and second common platform-independent processes to transmit requests thereto for information regarding one or more components of the SAN

(Bates' design allows for the storage allocator to communicate with servers and storage to process platform-specific processes (column 13, line 29- column 14, line 60, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not

specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

4. With regards to claims 5 and 24, Bates teaches through Phillips, the SAN, wherein the first and second common platform independent processes respond to the requests from the manager by invoking the first and second platform-specific processes, respectively

(Bates' design allows for the storage allocator to communicate with servers and storage to process platform-specific processes (column 13, line 29- column 14, line 60, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips).

Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

5. With regards to claims 6 and 25, Bates teaches through Phillips, the SAN, wherein the invoked first and second platform specific processes gather information regarding one or more SAN components and transmit the information to the Standard Output/Error of their respective first and second digital data processors

(Bates' design allows for the I/O (column 3, lines 37-41, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

6. With regards to claims 7 and 17, Bates teaches through Phillips, the SAN, wherein the first and second common platform independent processes capture information in the Standard Output/Error transmitted by the invoked first and second platform specific process

(Bates' design allows for the I/O (column 3, lines 37-41, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

7. With regards to claim 8, Bates teaches through Phillips, the SAN, wherein the common platform independent processes transmit the captured information to the manager for further processing

(Bates' design allows for the storage allocator to communicate with servers and storage to process platform-specific processes (column 13, line 29- column 14, line 60, Bates). While Bates teaches a SAN design using

different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

8. With regards to claims 9, 18 and 26, Bates teaches through Phillips, the SAN, wherein the manager comprises a query engine for transmitting the requests to the first and second common platform independent processes

(Bates' design allows query languages to be processed (column 15, lines 5-22, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time

of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

9. With regards to claims 10 and 19, Bates teaches through Phillips, the SAN, wherein the query engine comprises a registry identifying the first and second common platform independent processes and the first and second digital data processors, respectively, associated therewith

(Bates' design makes use of LUNs (equivalent to registry) (column 3, lines 25-34, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

10. With regards to claims 11 and 20, Bates teaches through Phillips, the SAN, wherein the registry provides one or more identifiers for communicating with the first and second common platform independent processes

(Bates' design makes use of LUNs (equivalent to registry) (column 3, lines 25-34, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

11. With regards to claim 12, Bates teaches through Phillips, the SAN, wherein the query engine formats the request in a mark-up language format

(Bates' design makes use various languages, including mark-up languages (column 15, lines 5-22, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations,

Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

12. With regards to claim 13, Bates teaches through Phillips, the SAN, wherein the mark-up language can be any of XML and HTML

(Bates' design makes use various languages, including mark-up languages (column 15, lines 5-22, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of

Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

13. With regards to claim 14, Bates teaches through Phillips, the SAN, wherein the platform independent processes format the captured information in a mark-up language format for transmission to the manager

(Bates' design makes use various languages, including mark-up languages (column 15, lines 5-22, Bates). While Bates teaches a SAN design using different operating systems (Figure 7, Bates), multiple processors (column 3, lines 46-67, Bates) and platform-specific operations, Bates does not specifically disclose the use of a command line interface between platform-specific processes.

Phillips teaches a method by which platform-specific processes are able to communicate via a command line interface (column 23, lines 50-67, Phillips). Therefore, it would have been obvious to one skilled in the art, during the time of the invention, to have combined the teachings of Bates with those of Phillips, to provide an interface between host servers and storage (column 8, lines 52-53, Bates)).

Remarks

The amendment received on April 25, 2006 has been carefully examined but is not deemed fully persuasive. The following are the examiner's response to the arguments presented within the latest amendment.

The first point of contention involves the cited portion of the Phillips prior art in teaching invoking and communicating with first and second command line interfaces of the first and second operating systems to effect execution of first and second platform specific processes via command line parameters. Applicant suggests that the prior art maintains a command line interface and does not use one. The prior art teaches how GDB (Unix based) communicates with Windows applications via ASCII in command line windows, which are retained (column 23, lines 50-67, Phillips). It is clear in the prior art that GDB and Windows are communicating through ASCII and it is extremely well known in the art that when command line windows can be left open as is in the prior art.

Another point of contention involves Figure 7 of the Bates prior art. The applicant suggests that Figure 7's element 704 fails to teach first and second platform independent processes using command line interfaces of the first and second operating systems, respectively. First, the Phillips prior art is applied to teach how the use of command line interfaces is known (in addition to the fact that it is well known in the art). Second, the Bates art's Figure 7 illustrates a SAN with first and second platform independent processes (Unix and NT in this case).

A further point of contention involves another cited portion of Bates. This time it is column 13, line 29 – column 14, line 60. Applicant yet again suggests that the cited portions fails to teach first and second platform independent processes using command

line interfaces of the first and second operating systems, respectively. The cited portions teach data transfer and storage features of storage area networks. The cited portions are directed towards the claim features of querying/requesting of data from the SAN.

The applicant continuously suggests that individually cited portions of the prior arts do not teach the first and second platform independent processes using command line interfaces of the first and second operating systems, respectively. However, the examiner provided a 103-type rejection and the applicant is reminded that the arts must be viewed in combination in order to properly understand the office action rejections.

The point of contention involves the feature of a query engine for transmitting requests to first and second common platform independent processes on different processors having different operating systems. First, Bates teaches the data request/querying means within column 13, line 29 – column 14, line 60. In addition, Figure 7 illustrates that different operating systems are able to make requests (i.e. element 702 is a NT server and elements 704 and 706 are Unix servers).

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not

mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Azizul Choudhury whose telephone number is (571) 272-3909. The examiner can normally be reached on M-F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jason Cardone can be reached on (571) 272-3933. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

AC

JASON CARDONE
SUPERVISORY PATENT EXAMINER